

Urszula KRĘGLEWSKA, Krzysztof SACHA\*

## WPROWADZENIE DO SYSTEMÓW CZASU RZECZYWISTEGO NA STUDIACH INFORMATYCZNYCH

Dydaktyka systemów czasu rzeczywistego jest na ogół prowadzona na wyższych semestrach studiów politechnicznych na wydziałach elektroniki lub informatyki, przy czym doświadczenie dowodzi, że studenci elektroniki łatwiej przyswajają idee pracy w czasie rzeczywistym od studentów informatyki. Kluczowym elementem problemu jest konieczność ścisłej współpracy systemu z instalacją zewnętrzną — przyszli informatycy, wychowani na problemach wirtualnej rzeczywistości, trudniej akceptują twarde ograniczenia narzucone na działanie oprogramowania przez potrzeby sterowanej instalacji. Przedmiotem tego artykułu jest laboratorium przedmiotu o nazwie „podstawy sterowania”, którego głównym zadaniem jest zapoznanie studentów z programowaniem aplikacji sterujących realnymi urządzeniami w czasie rzeczywistym. Przedmiot jest usytuowany we wczesnym etapie procesu dydaktycznego, na czwartym semestrze studiów inżynierskich na kierunku Informatyka. Na tym etapie studiowania nie można korzystać z żadnych zaawansowanych pojęć: ani z zakresu informatyki, np. systemów operacyjnych, ani teorii sterowania. Celem przedmiotu jest więc raczej przedstawienie studentom skąd się biorą problemy czasu rzeczywistego i jak ograniczają projektanta takiego systemu, niż nauczenie ich zaawansowanych metod rozwiązywania tych problemów. Na tę wiedzę, przyjdzie czas później.

### 1. WPROWADZENIE

Dydaktyka systemów czasu rzeczywistego jest na ogół prowadzona na wyższych semestrach studiów politechnicznych na wydziałach elektroniki. Wyodrębnienie informatyki jako samodzielnego kierunku studiów, często prowadzonego na osobnym wydziale, zrodziło potrzebę dostosowania dydaktyki systemów czasu rzeczywistego do innego profilu przygotowania studentów. Problem jest poważny — zgodne doświadczenia zaangażowanych w tę działalność wykładowców [2] pokazały, że studenci elektroniki (*electrical engineering*) znacznie łatwiej przyswajają idee pracy w czasie rzeczywistym od wychowanych na wirtualnej rzeczywistości studentów informatyki.

---

\* Instytut Automatyki i Informatyki Stosowanej Politechniki Warszawskiej,  
ul. Nowowiejska 15/19, 00-665 Warszawa.

Działania, podjęte w związku z tymi spostrzeżeniami na Wydziale Elektrotechniki Politechniki Warszawskiej po wyodrębnieniu kierunku Informatyka, doprowadziły do opracowania przedmiotu zapoznającego studentów z systemami czasu rzeczywistego już pod koniec drugiego roku czteroletnich studiów inżynierskich. Dalsze kształcenie w tym zakresie jest oferowane w ramach specjalizacji począwszy od semestru szóstego. Oferta specjalizacyjna obejmuje praktyczne przedmioty z zakresu inżynierii oprogramowania i programowania sterowników PLC oraz bardziej teoretyczne zagadnienia z zakresu algorytmów sterowania i optymalizacji oraz z dziedziny robotyki. Zaawansowane metody formalne modelowania systemów czasu rzeczywistego (sieci Petriego, języki typu SDL) wchodzi w skład programu dopiero na studiach magisterskich.

Zgodnie z definicją zamieszczoną w słowniku [3], system czasu rzeczywistego (*real-time system*) jest to system komputerowy, w którym obliczenia prowadzone równoległe z przebiegiem zewnętrznego procesu mają na celu nadzorowanie, sterowanie lub terminowe reagowanie na zachodzące w tym procesie zdarzenia. Definicja ta akcentuje dwa elementy wyróżniające systemy czasu rzeczywistego w olbrzymiej klasie systemów komputerowych:

- ściśle połączenie z zewnętrznym procesem (sterowaną instalacją), przebiegającym zgodnie z własną logiką wynikającą najczęściej z praw fizycznych,
- uzależnienia czasowe wyznaczające rytm działania systemu.

Ścisły związek systemu czasu rzeczywistego ze współpracującą z nim instalacją sprawia, że szczególnego znaczenia w początkowym etapie procesu dydaktycznego nabiera laboratorium, które musi dostarczyć studentowi niezbędnych intuicji. Równocześnie, ćwiczenia laboratoryjne usytuowane w pierwszej połowie studiów nie mogą korzystać z żadnych zaawansowanych pojęć: ani z zakresu informatyki, np. systemów operacyjnych, ani teorii sterowania.

Przedmiotem tego artykułu jest laboratorium przedmiotu „podstawy sterowania”, którego głównym zadaniem jest zapoznanie studentów z programowaniem aplikacji sterujących realnymi urządzeniami w czasie rzeczywistym. Przedmiot jest prowadzony na czwartym semestrze studiów inżynierskich na kierunku Informatyka. Przygotowanie studentów obejmuje, poza matematyką, dwu semestralny kurs podstaw elektroniki i trzy semestry nauki programowania. Związany z laboratorium wykład wprowadza studentów w podstawowe zagadnienia sterowania, optymalizacji i architektury systemów sterujących. Treść artykułu jest podzielona następująco. Rozdział drugi opisuje założenia, układ i następstwo ćwiczeń laboratoryjnych. Dwa następne rozdziały przedstawiają przykładowe urządzenia (instalacje) sterowane przez studentów w toku ćwiczeń. Ostatnim rozdziałem jest podsumowanie doświadczeń rocznej działalności laboratorium.

## 2. ZAKRES I ORGANIZACJA LABORATORIUM

Dziedzina systemów czasu rzeczywistego, widziana oczyma początkującego studenta informatyki, zawiera w sobie szereg osobliwości i trudności nie spotykanych w innych działach informatyki. Najważniejszym elementem jest tu konieczność poznania,

a następnie ściśle dostosowania biegu programu do trybu pracy, ograniczeń i wymagań zewnętrznych urządzeń. Tak silne uzależnienia nie występują w żadnych innych programach, z którymi student stykał się podczas kursu programowania. W zastosowaniach bliskim przemysłowej rzeczywistości występuje kolejny problem poznania specjalizowanych systemów, takich jak sterowniki PLC, regulatory i stacje operatorskie. Wszystkie wymienione systemy należą do grupy urządzeń programowanych, jednak języki i narzędzia programowania odbiegają wyraźnie od typowych języków i narzędzi informatycznych.

Pierwszym założeniem przyjętym przy opracowaniu programu laboratorium było ograniczenie trudności, z jakimi styka się student podczas każdego ćwiczenia, do jednego elementu. Inaczej mówiąc, każde ćwiczenie powinno dodawać jeden nowy element do już zgromadzonej wiedzy i umiejętności. Drugie założenie dotyczyło nieuchronnego kompromisu między „szerokością”, a „głębokością” kształcenia. Pierwsze laboratorium (pierwszy przedmiot z tego zakresu) powinno wprowadzić studenta w całość dziedziny aplikacyjnej, kosztem szczegółowości materiału. Trzecie założenie miało inny charakter i wynikało z twardych realiów ekonomicznych. Masowe laboratorium, przewidziane dla dwustu pięćdziesięciu studentów rocznie, musi być tanie, tzn. zbudowane z niedroгих, seryjnych elementów, tam gdzie to możliwe — ogólnego przeznaczenia.

Procesy przebiegające w instalacjach przemysłowych, transportowych lub komunalnych można podzielić na ciągłe i dyskretne. Kryterium tej klasyfikacji jest charakter sygnałów charakteryzujących bieżący stan procesu i sygnałów sterujących przebiegiem tego procesu. Najczęstszym przypadkiem sterowania dyskretnego jest sterowanie binarne, w którym wszystkie sygnały procesowe mogą przyjmować tylko jedną z dwóch wartości. Najczęstszym przypadkiem sterowania ciągłego jest regulacja — stabilizacja na wartości zadanej lub regulacja nadażna. Realne instalacje, zwłaszcza duże, zawierają na ogół elementy obydwu rodzajów. W ślad za tym system sterujący też musi zawierać urządzenia obydwu rodzajów, tworzące warstwę sterowania i regulacji. Ponad tą warstwą znajdują się urządzenia koordynujące i nadzorujące działanie warstwy sterowania bezpośredniego.

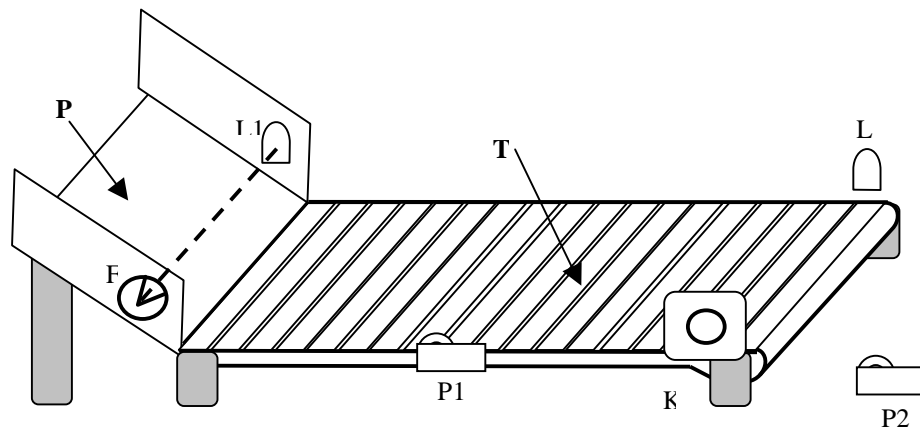
Zgodnie z założeniem kompletności, laboratorium powinno pokazać typowe problemy występujące we wszystkich wymienionych rodzajach zadań i warstwach systemu. Zgodnie z założeniem rozdzielenia trudności, każde ćwiczenie powinno jednak koncentrować się na jednym, dobrze określonym nowym elemencie. W tym celu całość programu została podzielona na pięć ćwiczeń:

1. Sterowanie binarne za pomocą standardowego komputera PC. Zadaniem studenta jest poznanie instalacji sterowanej, jej wymagań i ograniczeń, a następnie napisanie programu sterującego w dobrze mu znanym języku C lub C++.
2. Sterowanie binarne za pomocą sterownika PLC. Zadaniem studenta jest opracowanie programu sterującego w nowym i egzotycznym dla niego języku drabinkowym. Instalacją sterowaną jest tu instalacja poznana w trakcie poprzedniego ćwiczenia.
3. Stabilizacja poziomu cieczy za pomocą regulatora PID. Problem regulacji jest szczegółowo omawiany podczas wykładu przedmiotu. Przedmiotem ćwiczenia jest poznanie i skonfigurowanie regulatora przemysłowego.

4. Regulacja nadążna za pomocą standardowego komputera PC. W tym ćwiczeniu student poznaje sposób implementacji algorytmu sterującego i punkt widzenia projektanta urządzeń sterujących.
  5. Konfiguracja stacji operatora procesu. Zadaniem studenta jest przyjęcie informacji napływającej od urządzeń sterowania bezpośredniego i zaprojektowanie sposobu zaprezentowania tej informacji dla człowieka nadzorującego przebieg procesu.
- Nominalny wymiar laboratorium wynosi 15 godzin. Przytoczony wyżej program jest realizowany w postaci pięciu trzygodzinnych ćwiczeń.

## STEROWANIE BINARNE

Wyposażenie stanowiska składa się z komputera PC z kartą wejść-wyjść binarnych, napisanej dla potrzeb laboratorium biblioteki funkcji wejścia wyjścia i pomiaru czasu oraz obiektu laboratoryjnego z układem adaptera sygnałów. W drugim ćwiczeniu komputer jest zastępowany przez mały sterownik PLC Siemens-S7. Najciekawszym elementem wyposażenia są oczywiście obiekty, które zostały zbudowane z zabawkowych zestawów klocków technicznych. W dalszym ciągu rozdziału zostaną opisane dwa przykładowe obiekty.



Rys. 1. Linia rozdzielająca

Fig 1. Separating line

**Linia rozdzielająca** służy do rozdzielania elementów zawierających magnes od elementów bez magnesu. Przedstawiony na rys. 1 model linii jest zbudowany z pochylni **P**, po której klocki zsuwają się na taśmociąg **T** napędzany silnikiem oraz zespołu czujników i sygnalizatorów. W dolnej części pochylni znajduje się bariera świetlna złożona z żaróweczki **L1** i fototranzystora **F**. Czujnik **F** sygnalizuje przerwanie przez zsuwający się po pochylni obiekt promienia świetlnego emitowanego przez **L1**. Na końcu taśmociągu umieszczono czujnik **K** (kontaktron) wykrywający części magnetyczne

a także lampkę sygnalizacyjną **L2**. Dwa czujniki elektromechaniczne - przyciski **P1** i **P2** mogą być wykorzystane do sygnalizacji wybranych stanów linii sortującej. Rola czujników **F**, **K** i lampki **L1** w funkcjonowaniu linii sortującej jest jednoznaczna, natomiast funkcja pozostałych czujników i sygnalizatora **L2** może zależeć od postawionego w ćwiczeniu zadania.

Zadaniem studenta jest napisanie (w języku C lub w języku drabinkowym) programu realizującego zadany scenariusz działania linii. Na przykład:

„Naciśnięcie przycisku **P2** jest sygnałem do włączenia urządzenia. Należy wówczas zapalić lampkę **L1** i czekać aż wrzucony na pochylnię klocek przekroczy barierę świetlną. Sygnał z czujnika **F** powinien uruchomić taśmociąg, który przetransportuje klocek w pobliżu czujnika **K**. Jeśli klocek zawiera magnes, to pojawi się sygnał z czujnika **K**. Należy wówczas zatrzymać taśmociąg i zapalić lampkę **L2** na znak, że operator ma zdjąć klocek z taśmociągu. Taśmociąg rusza ponownie po otrzymaniu sygnału z czujnika **P2**. Powinna też zgasnąć wtedy lampka **L2**. Brak sygnału z czujnika **K** oznacza, że na taśmociągu są zwykłe klocki, które mają swobodnie spadać do stojącego pod taśmociągiem pojemnika. Taśmociąg będzie transportował klocki bez zatrzymywania do momentu pojawienia się klocka z magnesem w pobliżu czujnika **K**. Przełącznik **P1** służy do sygnalizacji konieczności awaryjnego zatrzymania taśmociągu. Ponowne uruchomienie układu następuje w taki sam sposób, jak odbywa się to przy początkowym włączeniu urządzenia.”

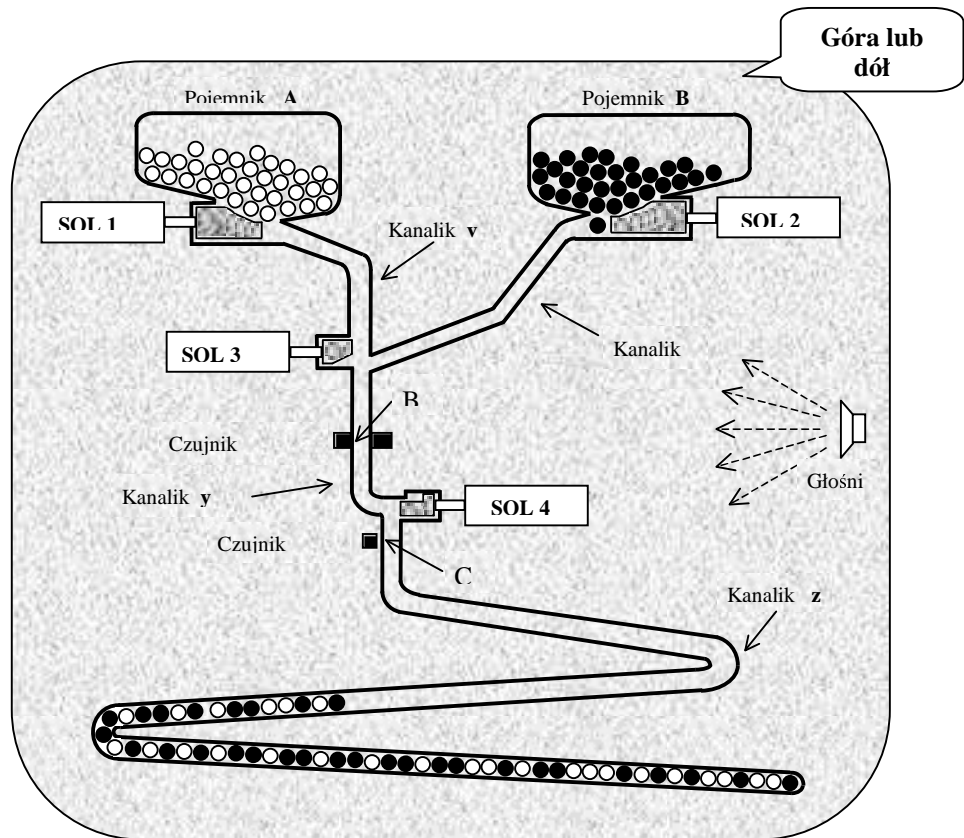
**Sorter kulek** może służyć do posortowania kulek różniących się kolorem lub do poukładania dwukolorowych kulek w określonym porządku. Urządzenie przedstawione na rys. 2 składa się z następujących elementów:

- dwóch pojemników **A** i **B**, w których mieści się wiele białych i czarnych kulek,
- kanalików **v**, **x**, i **y**, którymi kulki przesuwają się pod wpływem własnego ciężaru,
- zamkniętego na końcu kanalika **z**, w którym kulki układają się w pojedynczy sznur,
- optycznego czujnika **BS**, sygnalizującego fakt przejścia kulki przez punkt **BS**,
- optycznego czujnika **CS** koloru kulki przechodzącej przez punkt **CS**,
- czterech bramek elektromagnetycznych **SOL1**, **SOL2**, **SOL3** i **SOL4** zamykających kanaliki w określonych miejscach,
- sygnalizatora dźwiękowego – głośnika.

Urządzenie można ustawiać w dwóch położeniach. W pierwszym położeniu zbiorniki **A** i **B** znajdują się na górze, a kanalik **z** na dole. Dzięki temu kulki przemieszczając się z góry na dół, mogą po opuszczeniu pojemników **A** i **B** formować sznur w końcowym kanaliku **z**. W drugim położeniu kanalik **z** znajduje się na górze, a zbiorniki **A** i **B** na dole. Kulki staczające się z kanalika **z** są rozdzielane do dwóch pojemników. W zależności od wystawienia cewek elektromagnesów **SOL1**, **SOL2**, **SOL3** i **SOL4** w trakcie opadania kulek uzyskuje się różne ich rozłożenie w pojemnikach **A** i **B** albo kanaliku końcowym **z**.

Istotną rolę w sterowaniu cewek odgrywają czujniki obecności i koloru kulki, przy czym każdy z nich ma znaczenie dla innego wariantu działania urządzenia. I tak czujnik obecności kulki jest wykorzystywany przy mieszaniu kulek wypuszczanych z pojemnika kulek białych i pojemnika kulek czarnych. Po wypuszczeniu kulki czarnej z pojemnika **A**

można bezpiecznie otworzyć drogę dla kulki białej dopiero po uzyskaniu informacji, że poprzednia kulka przeleciała przez obszar **BS**. Natomiast czujnik koloru kulki jest przydatny podczas operacji rozdzielania kulek znajdujących się w kanaliku końcowym **z**, do dwóch zbiorników w zależności od koloru kulek.



Rys. 2. Sorter kulek  
Fig 2. Sorter

Zadaniem studenta jest napisanie (w języku C lub w języku drabinkowym) programu realizującego zadany algorytm działania sortera. Na przykład, algorytm rozdzielania kulek znajdujących się w kanaliku **z** do zbiorników **A** (kulki białe) i **B** (kulki czarne) można zapisać następująco:

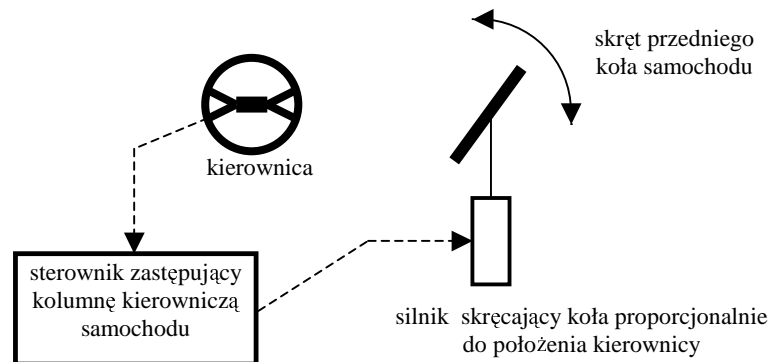
„Początkowo panel jest ustawiony zbiornikami do góry. Żadna z cewek nie jest włączona i wszystkie kulki mogą swobodnie przelecieć z pojemników **A** i **B** do kanalika **z**. W operacji rozdzielania kulek biorą udział cewki **SOL4** i **SOL3** oraz czujnik koloru **CS**. Aby wykonać operację rozdzielania należy najpierw wysteroować cewkę **SOL4** (aby zamknąć drogę dla kulek), a następnie obrócić panel tak, aby u góry znalazł się kanalik **z**. Sznur kulek przemieści się trochę w kierunku pojemników i zostanie zatrzymany w punkcie **CS**.

Sygnal z czujnika koloru pozwoli określić kolor kulki znajdującej się najbliżej bramki zamykanej przez **SOL4**. Jeżeli kulka jest biała, to w trakcie jej przelotu przez kanałik **y** nie zostanie wysterowana cewka **SOL3** i kulka swobodnie przemieści się do pojemnika **A**. Jeżeli kulka jest czarna, to na czas jej przelotu przez kanałik **y** należy wysterować cewkę **SOL3**, która zagrozi drogę do pojemnika **A** i skieruje kulkę do pojemnika **B**.

Kiedy już droga dla kulki zostanie wytyczona stosownie do jej koloru, należy na chwilę wyłączyć cewkę **SOL4**, aby przeleciała jedna kulka. Czas wyłączenia cewki **SOL4** nie może być za długi, aby przeleciała tylko ta jedna kulka, której kolor jest znany. Długość tego czasu wynosi około 200 ms, ale dokładną wartość trzeba wyznaczyć metodą prób. Przejście kulki przez cały kanałik **y** i rozgałęzienie **v/x** wymaga trochę czasu (około 300 ms), po którym dopiero można, po uprzednim rozpoznaniu kolejnej kulki, otworzyć dla niej drogę. Przepuszczanie i rozdzielanie kolejnych kulek odbywa się tak samo, poczynając od ustalenia koloru kulki. Należy uwzględnić niezerowy czas identyfikacji koloru kulki i pozostawić przerwę 50 ms przed dokonaniem kolejnego odczytu z czujnika koloru.

### 3. REGULACJA

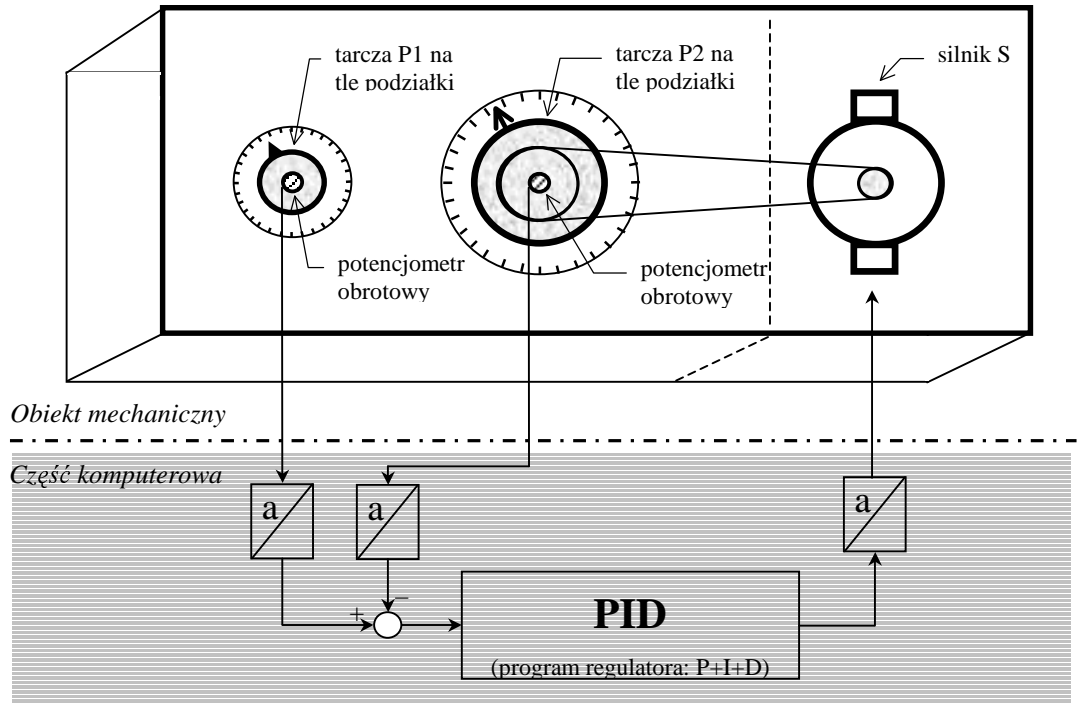
Wyposażenie stanowiska składa się z komputera PC z kartą wejść-wyjść analogowych, napisanej dla potrzeb laboratorium biblioteki funkcji wejścia wyjścia i pomiaru czasu oraz obiektu laboratoryjnego połączonego z komputerem poprzez układ adaptera sygnałów. Najciekawszym elementem wyposażenia jest obiekt regulacji, wykonany indywidualnie dla potrzeb tego ćwiczenia. Urządzenie to może być modelem serwomechanizmu kierowania samochodem, opracowywanego dla przemysłu motoryzacyjnego i przedstawionego na rys. 3. Zadanie studentów podczas ćwiczenia polega na napisaniu programu sterującego, w taki sposób aby tarcza P2 modelująca koła nadążała za zmianami pozycji tarczy P1 modelującej kierownicę samochodu.



Rys. 3. Przykładowa instalacja sterowana: serwomechanizm kierowania samochodem

Fig 3. Drive-by-wire car control example

Rzeczywista budowa obiektu jest pokazana na rys. 4. Obiekt składa się z poruszanej ręką tarczy zadajnika położenia P1 (odpowiednik kierownicy) oraz tarczy P2 (odpowiednik przednich kół samochodu) obracanej przez silnik elektryczny S. Pomiar położenia kąowego zadajnika P1 oraz położenia kąowego tarczy P2 jest oparty na sprzężeniu ich z wieloobrotowymi potencjometrami precyzyjnymi, zasilanymi napięciem stałym. Napięcie na suwaku potencjometru, odczytywane przez komputer, odwzorowuje bieżące położenie tarczy.



Rys. 4. Serwomechanizm laboratoryjny

Fig 4. Control rig

Konstrukcja mechaniczna urządzenia umożliwia wykonanie prawie 2 obrotów tarczy zadajnika P1. Podobnie tarcza sprzężona z silnikiem może wykonać prawie 2 obroty. Krańcowe położenia tarczy poruszanej przez silnik są chronione elektronicznie i nie mogą być przekroczone w wyniku niewłaściwego sterowania przez komputer. Poza obwodem tarczy jest widoczna biegnąca wokół tarczy podziałka, złożona z 72 kresk. Przesunięcie wskazówki zadajnika lub wybranego punktu na tarczy od kreski do kreski oznacza obrót kątowy o 5°.



Opisane w tym artykule laboratorium jest prowadzone co semestr, od początku bieżącego roku akademickiego. Istnieją również wcześniejsze doświadczenia związane z eksperymentalnymi zajęciami wykorzystującymi sukcesywnie powstające stanowiska. Zebrane opinie — osób prowadzących ćwiczenia i studentów — potwierdzają słuszność przyjętych założeń i trafność wyboru metod ich realizacji.

Studenci oceniają laboratorium jako dość trudne ale rozwijające, a przypadku dobrych studentów także dostarczające dużo zabawy. Dobrze skalkulowany okazał się czas trwania ćwiczeń — 3 godziny lekcyjne. Tylko sporadycznie pojawiają się zespoły kończące swoje zadanie przed upływem 2,5 godziny, a jednocześnie rzadkie są przypadki nie zaliczenia ćwiczenia.

Dokumentacja laboratoryjna obejmuje pisemne instrukcje do wszystkich ćwiczeń. Każda instrukcja zawiera wprowadzenie w dziedzinę aplikacji, opis obiektu laboratoryjnego, opis biblioteki funkcji programistycznych lub języka programowania (sterownika lub konfiguratora regulatora) oraz scenariusz przykładowego ćwiczenia. W najbliższym czasie ukaże się skrypt zawierający pełną wersję instrukcji [1].

#### LITERATURA

- [1] KRĘGLEWSKA U. (ed.), Laboratorium podstaw sterowania, Oficyna Wydawnicza PW, w druku.
- [2] NAWROCKI J., SCHWARZ J.-J., ZALEWSKI J. (eds), Real-Time Systems Education III, IEEE Computer Society, 1998.
- [3] Standard Computer Dictionary, IEEE Std 610, 1990.

### **INTRODUCTION TO REAL TIME SYSTEMS EDUCATION**

It is a challenging task to develop an introductory course in real-time control systems which includes a laboratory in which to experiment the concepts. The challenge is made even more difficult when the course is offered in the middle of the undergraduate studies, when the theoretical background of the students is poor and the funds for such an endeavor are scarce. The goal of this paper is to report on introductory lab course on real-time systems at Warsaw University of Technology. Practical projects were implemented which give students hands-on experience with real or simulated mechanical and electronic devices while learning the software requirements and design constraints of real-time systems. We discuss in the paper the setup of the course in relation to other courses and briefly discuss some of our cases used in the lab. Some conclusions based on our experiences are presented.